

Amendments to Claims

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims

1. (original) A method for accelerating access to data on a network comprising:
providing a plurality of computers on the network, each with cache software;
receiving in one of the computers, from an application, a write instruction that
specifically addresses data in a cached I/O device connected to the network;
maintaining exclusive write access control over the specifically addressed data in the
cached I/O device;
writing data into cache in the one of the computers responsive to the write instruction;
communicating over the network to invalidate data in remote caches on the network
that cache the cached I/O device;
sending a write I/O completion signal to the application after completing the
invalidation of the specifically addressed data in the remote caches on the network.
2. (original) The method of claim 1 wherein the specifically addressed data is a
single data block.
3. (original) The method of claim 1 further comprising releasing the exclusive
write access control after completing the invalidation of the specifically addressed data in the
remote caches on the network.
4. (original) A method for coherently caching a shared I/O device available on a
network comprising:
providing a plurality of computers on the network each with cache software; and

creating a data structure that provides a list of the computers on the network that permit caching with respect to the shared I/O device.

5. (original) The method of claim 4 further comprising when data is written into a local cache, communicating with each computer in the list of computers to invalidate data in remote caches corresponding to the shared I/O device.

6. (original) A method for accelerating access to data on a network comprising:
providing a plurality of computers on the network, each with cache software;
receiving a write instruction at one of the computers;
maintaining exclusive write access control over a single data block addressed in the write instruction;
writing data into cache in the one of the computers responsive to the write instruction;
communicating over the network to invalidate data in remote caches corresponding to the addressed single data block.

7. (currently amended) A method for ~~coherently caching I/O devices available for shared access on a network~~ accelerating access to data on a network comprising:

providing a plurality of computers on the network, each with cache software;
receiving, in one of the computers, a write instruction having data to be written into addresses on one of said I/O devices;

selecting one of a plurality of cache data bucket sizes available in the one of the computers;

writing the data into a data bucket of the selected data bucket size responsive to the write instruction; and

communicating over the network with remote caches to invalidate cache ~~coherently~~
~~caching an I/O device available for shared access on a network~~ data corresponding to any of
the addresses.

8. (original) The method of claim 7 wherein a byte count of the data written
responsive to the write instruction determines the cache data bucket size selected.

9. (original) A method for coherently caching an I/O device available for shared
access on a network comprising:

providing a plurality of computers on the network, each with cache software; and
privately communicating on a computer communication channel between cache
software on a first of the computers caching the I/O device and cache software on a second of
the computers caching the I/O device.

10. (original) The method of claim 9 further comprising intercepting, in the cache
software of the first of the computers, a write instruction to the I/O device.

11. (original) The method of claim 10 wherein privately communicating
comprises sending a message through the computer communication channel from the cache
software of the first of the computers to the cache software of the second of the computers to
invalidate data responsive to the write instruction.

12. (original) The method of claim 11 further comprising privately
communicating on computer communication channels with all of the computers caching the
I/O device to invalidate data responsive to the write instruction.

13. (original) A method for coherently caching I/O devices available for shared
access on a network comprising:

providing a plurality of computers on the network, each with cache software, the cache software for storing data into any of a plurality of caches each being of a different data bucket size;

receiving a write instruction having data to be written into addresses in one of the I/O devices;

writing data into one of the caches responsive to the write instruction; and

communicating over the network to invalidate cache data corresponding to addresses that overlap with addresses for the data in the write instruction.

14. (original) A method for coherently caching I/O devices available for shared access on a network comprising:

providing a network with a plurality of nodes, each with a cache driver;

creating a cache for one of the I/O devices at one of the nodes not currently caching said one of the I/O devices; and

communicating with all the cache drivers at all the nodes on the network that permit caching with respect to said one of the I/O devices to inform them that said one of the I/O devices may be cached on said one of the nodes.

15. (original) A method for coherently caching I/O devices available for shared access on a network comprising:

providing a network with a plurality of nodes, each with a cache driver; and

registering each cached I/O device with all the cache drivers on all nodes that cache said cached I/O device.

16. (original) The method of claim 15 further comprising forming communication channels with each of the nodes in said plurality of nodes.

17. (original) A method for coherently caching I/O devices available for shared access on a network, comprising:

providing a network with a plurality of nodes, each with a cache driver; and

creating, at each node that may cache one of the I/O devices, a data structure for the one of the I/O devices that includes a list of all nodes on said network that permit caching with respect to the one of the I/O devices.